

Integrality and Separability of Multi-touch Interaction Techniques in 3D Manipulation Tasks

Anthony Martinet, Géry Casiez, Laurent Grisoni
LIFL, INRIA Lille & University of Lille, FRANCE

Abstract—Multi-touch displays represent a promising technology for the display and manipulation of data. While the manipulation of 2D data has been widely explored, 3D manipulation with multi-touch displays remains largely unexplored. Based on an analysis of the integration and separation of degrees of freedom, we propose a taxonomy for 3D manipulation techniques with multi-touch displays. Using that taxonomy, we introduce *DS3* (*Depth-Separated Screen-Space*), a new 3D manipulation technique based on the separation of translation and rotation. In a controlled experiment, we compared *DS3* with *Sticky Tools* and *Screen-Space*. Results show that separating the control of translation and rotation significantly affects performance for 3D manipulation, with *DS3* performing faster than the two other techniques.

Index Terms—Multi-touch displays, 3D manipulation task, direct manipulation, DOF separation.

1 INTRODUCTION

Three-dimensional (3D) manipulation is a challenge for 3D interface designers since it involves the control of six Degrees Of Freedom (DOF) : three for position (i.e. translation along x , y and z -axis) and three for orientation (i.e. rotation around x , y and z -axis). Using a mouse with a current desktop interface, rotating 3D objects can take from ten to thirty seconds [1], much slower than real object manipulation which takes between one and two seconds [2].

Compared to the mouse, multi-touch displays provide extra input bandwidth through multiple contact points and enable direct manipulation allowing users to directly *touch* data [3]. The Rotate-Scale-Translation gesture (RST) for manipulating 2D data is a typical example of this type of interaction paradigm [4]. While 2D manipulation on multi-touch displays has been widely explored, 3D manipulation has not. This may be explained by the mapping difficulty of the inherently 2D input contact points to the 3D attributes which need to be controlled.

Jacob *et al.* [5] studied the impact of the input device control structure and the perceptual structure on task performance, with a focus on whether the structures were integral or separable. They found a strong relationship between the two structures with better performance when both match. While it has been shown that human fingers have separable DOF [6], 3D manipulation is inherently an integral task [7]. The thumb, index and middle fingers can be moved

separately from one another while users perceive the attributes of 3D objects (position and orientation) as a whole. This mismatch between the separable input structure of multi-touch devices and integral nature of a 3D manipulation task raises the question of how to optimize the mapping between the two structures.

While techniques like *Sticky Tools* [8] propose a way to separate the DOF of a 3D manipulation task, other techniques like *Screen-Space* [9] present a method to integrate them. However, the lack of user-study makes it difficult to compare the two approaches.

After presenting the related work on DOF integration and separation, the existing metrics to measure DOF coordination, and the 3D manipulation techniques

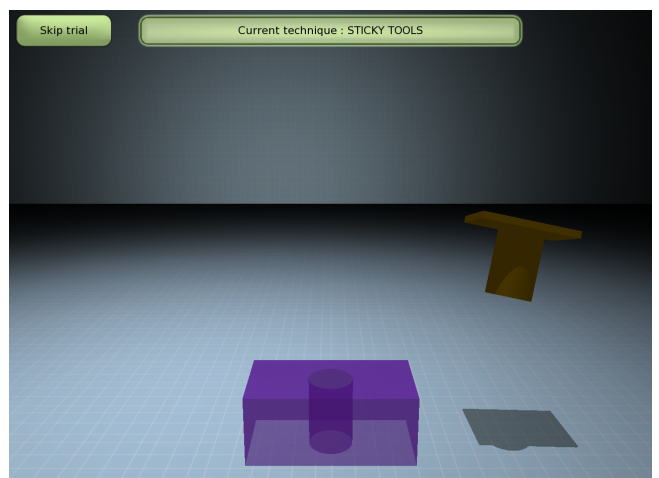


Fig. 1: Screen capture of the peg-in-hole task.

e-mail: anthony.martinet@lifl.fr
e-mail: gery.casiez@lifl.fr
e-mail: laurent.grisoni@lifl.fr

for multi-touch displays, we introduce a taxonomy to compare 3D manipulation techniques for multi-touch displays. Then, we introduce a new technique called *DS3* (*Depth-Separated Screen-Space*) based on the clear separation between translation and rotation, and present the results of a controlled experiment comparing this technique with *Sticky Tools* [8] and *Screen-Space* [9]. Finally, in the discussion we address the question of controlling a task with integrated DOF using a separable multi-touch input device.

2 RELATED WORK

2.1 Integration and Separation of DOF

According to the theory of the perceptual structure of visual information by Garner [7], a multi-dimensional object can be characterized by its attributes in two categories: integral structure and separable structure. Visual information has an integral structure if its attributes can be perceptually combined to form a unitary whole. If visual object attributes show perceptually distinct and identifiable dimensions, they are separable. According to this definition, the orientation and the position of a 3D object are two integral attributes, making 3D manipulation an integral task.

Jacob *et al.* [5] extended Garner's notion of integral and separable structure to interactive tasks by observing that manipulating a graphic object is simply the modification of the values of its attributes. They also extended integral and separable structure to describe the attributes of an input device, based on whether it is natural to move diagonally across all dimensions. With an integral device, the movement is in Euclidean space and cuts across all the dimensions of control. A separable device constrains movement along one dimension at a time. They conducted an experiment in which participants performed two tasks that had different perceptual structures, using two input devices with correspondingly different control structures: an integral three-dimensional tracker and a separable mouse. Their results support their hypothesis: human performance increases when the perceptual structure of the task matches the control structure of the device. They concluded that the interplay between task and device was more important in determining performance than either task or device alone.

Wang *et al.* [2] extended this theory to extrinsic (i.e. orientation and position) properties of an object being manipulated by the human hand. They pointed out that human visual pathways (i.e. human processing chains) responsible for object perception are separated from those guiding the action. They ran an experiment that asked participants to dock a real wood cube using different visual feedback conditions. They reported that users had little difficulty in simultaneous control of object translation and orientation.

Considering an orientation task only, Veit *et al.* [10] studied the integration of DOF. They conducted an experiment in which users had to orient 3D objects using two interaction techniques, one integrating and the other separating the DOF of the orientation task. The results suggest that the simultaneous manipulation of all the DOF does not necessarily lead to the best performance, leading to conclusions opposite to those of Jacob *et al.*

Regarding 2D manipulation with multi-touch displays, Nacenta *et al.* [11] addressed the issue of manipulating a given subset of DOF. When using multi-touch gestures, performing a subset of available operations may be difficult for users. For example, it can be hard to only scale and translate an object (without rotating it) because the object will also react to small variations of the angle between the contact points. They introduced an interaction technique that allows users to select a subset of DOF, reducing unwanted manipulation without negatively affecting performance. Separating the control of DOF like this improved the user's expectations.

2.2 Metrics for Quantifying DOF Coordination

While the concepts of integration and separation of DOF are useful to describe the attributes of tasks and input devices from a theoretical point of view, some metrics have been introduced to actually measure DOF coordination.

To measure the degree of DOF coordination for translation, Jacob *et al.* [5] first segmented the movement trajectory into equal units of time. Using distance thresholds, each time segment was then classified as Euclidean if the corresponding trajectory showed movement in more than one dimension or city-block if the trajectory only occurred along a single dimension. The degree of coordination was then defined as the ratio of Euclidean to city-block movements. However this metric is not defined when the number of city-block movements is equal to zero. The metric also does not provide any baseline for perfect coordination nor does it distinguish between movements which contribute towards the goal and movements which do not.

Zhai *et al.* proposed another metric for measuring DOF coordination in docking tasks [12]. The translation coordination is computed as the ratio of the lengths of the shortest path to the actual path. The rotation coordination is computed in a similar way using the ratio of the initial rotation mismatch to the amount of actual rotation. A ratio close to one corresponds to a high degree of coordination between the DOF of the input device while a ratio close to zero corresponds to a poor coordination between them.

The translation and rotation coordination values provide a global measure of coordination independent of the temporal profile of the movement trajectory. As a result, it is not possible to know when the rotation and translation occurred as it is not possible to determine if translation and rotation occurred simultaneously. To address this, Zhai simply used a 2D visual representation [12] of the translation and rotation over time, where the x-axis represents the translation coordination and the y-axis represents the rotation coordination. We will refer to this metric as the *translation-rotation* coordination ratio. However this visual representation does not provide any numerical value to represent the simultaneous coordination of the translation and rotation DOF.

To solve this problem, Masliah *et al.* [13] introduced the *m*-metric. The goal of the *m*-metric is to compare a tuple of n DOF (where $n \geq 2$) in a docking task that require the manipulation of m DOF ($n < m$). The metric measures the degree of simultaneous error reduction occurring in multiple DOFs. It takes into account both the simultaneity (i.e. simultaneous manipulation of multiple DOF) and the efficiency of control (i.e. manipulation of DOF that reduce the error) across the DOF. Error is defined as the distance between the current position and the target position. In other words, the *m*-metric can be used to understand precisely which subsets of DOF are manipulated during a docking trial.

More recently, Veit *et al.* [14] introduced the *Number of Degrees of Freedom Combined* (NDC), a metric providing the number of DOF simultaneously manipulated during a multi-DOF docking task. They tested the metric on an orientation task and explained how it could be generalized for all 6 DOF tasks.

To sum up, Zhai's metric provides a global measure of the coordination while the NDC captures how many DOF are manipulated over time. Finally the *m*-metric details which subset of DOF are manipulated together. From a qualitative point-of-view, the visual representation introduced with the *translation-rotation* coordination ratio helps to understand users' strategies to reach the target position and orientation.

2.3 3D Manipulation with Multi-touch Displays

Limited 3D manipulation

Using a standard vision-based tabletop, Wilson *et al.* [15] created a physics-enabled 3D environment with multi-touch input manipulating the DOF of the task in an integral way. Their technique is able to model both multiple contact points and more sophisticated shape information, such as the entire hand. They showed that their technique can be used to add real-world dynamics to interactive surfaces. While the

underlying physical simulation can provide a number of convincing effects during the interaction (inertia, collision), integrating all the DOF in such an environment prevents users from lifting object (i.e. move the object along the z -axis).

Hilliges *et al.* [16] used depth-sensing camera to solve this issue so that users can "pick-up" an object and manipulate it above the surface. This supports Jacob's conclusions since extra input information improves the match between the control structure and the task perceived structure, both integral in this case. However, those techniques require additional hardware making the compatibility with existing multi-touch displays difficult.

Martinet *et al.* [17] proposed two techniques for 3D positioning. One technique, the *Z*-technique, presented 3D data in full-screen while the other technique split the screen in 4 viewports. They conducted a docking task experiment, but were not able to draw conclusions on performance. From a qualitative point of view, they reported that users preferred the full-screen technique.

Full 3D manipulation

Hancock *et al.* [18] presented one, two and three touch input interaction techniques to manipulate 3D objects on multi-touch displays. With three-touch interaction, users can perform simultaneous translation and rotation on the surface of the table. Depth-positioning is activated as an option, by measuring the distance between two fingers. The three-touch technique, called *Shallow-Depth*, was found to be faster and more accurate and also preferred by their users. Nevertheless, the 3D task used in the experiment only required the manipulation of 5 DOF. To control all 6 DOF required for 3D manipulation, they introduced the *Sticky Tools* technique [8], allowing users to manipulate an object using three fingers. Each finger separately controls DOF that are integrated together. While the authors discussed the use of such a technique in a more general manner, the choice of DOF to integrate together is not addressed and no user study was carried out to measure the efficiency of *Sticky Tools*.

Reisman *et al.* [9] introduced a method to handle 3D manipulation in a direct way, integrating all the DOF needed to perform such an operation. Highlighting the fact that RST has become the *de facto* standard technique to handle 2D objects, they presented a technique to extend RST into 3D. The tool works by solving constraints fixed by users' fingers. A constraint solver minimizes the error between the screen-space projection of contact points (i.e. finger positions on the 2D screen) and their corresponding screen-space target positions (i.e. the 3D points *touched* by fingers). The paper discusses the use of the constraint solver

and provides examples to use this tool to design interaction techniques — but no formal evaluation was performed.

Recently, Cohé et al. [19] introduced tBox, a technique that combines 3D widgets and direct manipulation. tBox separates all the DOF of the manipulation using the bounding box of an object. Scaling is also possible via this technique. However the technique was not formally evaluated nor compared.

3 A TAXONOMY OF 3D MANIPULATION TECHNIQUES WITH MULTI-TOUCH DISPLAYS

The 3D manipulation techniques mentioned above control different sub-sets of DOF depending on the number of fingers in contact with the surface. In addition a finger can be considered either *direct* or *indirect* depending on the euclidian physical distance between the finger position and the projection on screen of the virtual object being manipulated. When this distance is equal or close to zero, the finger is *direct* and turns *indirect* when this distance becomes greater. The number of fingers used for the interaction and the directness of each finger is referenced below as a mode.

To help comparing existing manipulation techniques, we chose to adapt the taxonomy introduced by Card *et al.* [20]. We wanted to represent the relationship between the number of fingers, their directness (whether *direct* or *indirect*) and the corresponding DOF controlled in the task. We also wanted to represent whether the DOF of the task is controlled in an integral or separable way. The DOF controlled in the manipulation task are represented in a cartesian direct framework where the x -axis belongs to the screen plane and is oriented towards the right and the z -axis is orthogonal to the screen and points towards the user. T_x , T_y and T_z represent the translations along the corresponding axis; R_x , R_y and R_z the rotations around the corresponding axis. This taxonomy only takes into account the information available through the inputs provided by the current technology: the number of contact points (e.g. the number of fingers) and the directness of each finger. This taxonomy could be enriched by specifying the name and associated hand for each finger in contact, and also the order in which they have to enter in contact with the surface.

In figure 2, we represent an illustration of the use of the taxonomy with *Sticky Tools*. Each line represents a mode for the technique, annotated with the number of fingers associated. Indirect fingers are represented with an “i” in the corresponding circles, whereas direct fingers are left blank. Circles connected together with a single-line represent the DOF of the task controlled in an integral way. Groups of circles

	Mode	Translation			Rotation		
		T_x	T_y	T_z	R_x	R_y	R_z
Sticky Tools	1d	○	○				
	2d	○	○	○			○
	1d + 1i	○	○		i	i	
	2d + 1i	○	○	○	i	i	○

Fig. 2: Description of the *Sticky Tools* technique using the taxonomy.

disconnected represent the DOF of the task controlled in a separable way.

For the *Sticky Tools* technique represented in figure 2, the mode 1d represents the first finger in contact with the object to manipulate, which controls the object translation along the screen plane in a direct and integral way. When a second finger enters in contact with the same object (mode 2d), translation and rotation around the z -axis are now possible in a direct and integral way, in addition to the DOF controlled by the first finger (i.e. each finger movement can now change four DOF at once). The second finger can also be used in an indirect way (mode 1d+1i) to control two DOF in rotation in an integral way but separately from the DOF controlled by the first finger. Last, the mode 2d+1i shows the combination of the previous modes to control the six degrees of freedom at the same time, but with two DOF in rotation being controlled separately.

4 SCREEN-SPACE TECHNIQUE

As mentioned previously, Reisman *et al.* [9] introduced a method to perform 3D manipulation with multi-touch displays. We refer to this technique as *Screen-Space*. This method uses a constrain solver to integrate all the DOF of the manipulation. The solver takes user’s fingers as input and returns the updated values of the DOF. The calculation is a least-squares minimization of the error between the screen-space projection of contact points and their corresponding screen-space target positions. A simplified version of the algorithm can be described as follows :

- 1) When a finger touches a 3D object projected on screen :
 - Record the 2D location of the finger on screen (point $F2d_1$)
 - Record the 3D point corresponding to the ray-casting of the finger 2D position into the 3D scene (point $P3d_1$)

	Mode	Translation			Rotation		
		Tx	Ty	Tz	Rx	Ry	Rz
Screen-Space	1d	○	○				
	2d	Added after pilot study			○	○	○
	$\geq 3d$	○	○	○	○	○	○

Fig. 3: Description of the *Screen-Space* technique using the taxonomy.

2) When a finger moves:

- Record the new position (point $F2d_2$)
- Use the constrain solver to adjust the position and orientation of the 3D object so that when $F2d_2$ is casted into the scene, it points to $P3d_1$

The goal of the algorithm is to match user's fingers to 3D points and keep these 3D points *stuck under* user's fingers when they move. When it comes to scale and rotate a 2D picture using multi-touch input, it is exactly the same process but instead of matching 2D points (i.e. fingers) with 3D points (i.e. 3D object), 2D points (i.e. fingers) are matched with 2D points (i.e. 2D picture).

To control the six DOF required for 3D manipulation, at least three fingers are required, as a single finger can only control two DOF at best (we consider here only the x,y positions of fingers). With less than three fingers, the interface designer has to choose the DOF controlled in the task. With one finger, the natural choice is to control the translation of the object in the camera plane, as illustrated with mode 1d in figure 3. With two fingers, up to four DOF among the six can be controlled. Reisman *et al.* do not recommend any particular mapping. The mode 2d presents one possible mapping chosen after a pilot study we discuss later.

5 INTRODUCING DS3

According to Garner [7], the perceptual structure of 3D manipulation consists of six integrated DOF. Jacob *et al.* [5] recommend matching the perceptual structure of the task with the control structure of the input device. Strictly following these two recommendations leads to using only input devices with six integrated DOF such as 3D mice or 3D wands, and to interact with the whole hand instead of only interacting with fingers.

Considering the separable structure of fingers [6], it appears impossible to exactly match the perceptual

structure of the task with the control structure of multi-touch displays. The previous work above shows there is no clear answer to this problem. On the one hand *Screen-Space* proposes to control the six degrees of freedom in an integral way and on the other hand *Sticky Tools* proposes a separation between the degrees of freedom. As these two techniques were not evaluated nor compared, it is difficult to know which approach is the best. If the DOF separation appears better, it also addresses the question of the best way to separate DOF.

During an informal evaluation of *Sticky Tools*, we observed that the integral control of translation and rotation (modes 2d and 2d+1i in figure 2) is indeed difficult. When DOF are controlled separately, our hypothesis is that a clear separation of translation and rotation improves user efficiency. As a consequence we designed a new technique clearly separating the control of rotation from the control of translation. We called this technique *DS3*. *DS3* combines the *Z*-technique [17] used to control the position; and the constrain solver described by Reisman *et al.* [9] to control the orientation.

With one direct finger, objects can be translated along the screen plane (mode 1d in figure 4). Depth translation is performed in an indirect way, with a second, indirect finger. When this finger is in contact with the surface we measure its relative motion on the surface and use backward and forward movement to control the depth position. Forward movement moves the object away from the user view and backward movement moves it closer to the user's view.

With at least two direct fingers, users can control only the orientation of the object in an integral way using the constrain solver previously described. Finger positions are used as inputs for the constrain solver, which provides us the appropriate rotation to perform¹.

The number of fingers directly in contact with the object (one vs. two or more) provides a clear separation between translation and rotation. In addition, when rotating the object, we also allow the manipulation of the object depth (i.e. translation along *z*-axis) with an indirect finger, as previously described. This is not a breach of the separation of position and orientation since depth-position is handled with a additional separated finger.

6 PILOT EXPERIMENT

This pilot experiment was designed to pre-test *Sticky Tools*, *Screen-Space* and *DS3* on a real task. We also wanted to examine different mappings for the mode

1. <http://www.youtube.com/watch?v=DnHvpyjgYik>

	Mode	Translation			Rotation		
		Tx	Ty	Tz	Rx	Ry	Rz
DS3	1d						
	1d + 1i						
	$\geq 2d$						
	$\geq 2d + 1i$						

Fig. 4: Description of the DS3 technique using the taxonomy.

2d of Screen-Space. In addition, this allowed us to tune technique parameters.

With *Screen-Space*, we can control up to 4 DOF with 2 fingers. To remove unintended translation as mentioned by Hancock *et al.* [18], we decided to remove the two DOF which were mapped to the one finger mode, leaving us with the four remaining DOF. Since we believe that separating rotation can improve efficiency, we decided to use two fingers mode for controlling rotation DOF only.

6.1 Apparatus

The experiment was conducted on an Immersion iLight² touch table based on the Diffused Illumination technique. The surface is a 100 cm × 70 cm (42 inches) monoscopic display positioned 105 cm above the floor. The video-projector under the table was set at 60 Hz with a 1400 × 1050 pixel resolution giving a pixel density of 14 pixels per cm (36 DPI). A camera running at 120 Hz with a 640 × 480 pixel resolution is positioned under the surface to capture finger movements. This gives a maximum resolution of 6.4 dots per cm (16.25 DPI) for finger tracking. We used the iLight framework version 1.6 for fingers detection and tracking. Finger data was then sent using TUIO messages³ to a custom built 3D application based on the Ogre3D framework⁴. The source code of the Ogre3D listener implementing the different interaction techniques is available on github⁵.

6.2 Task and Participants

The task is a 3D peg-in-hole task similar to the one described by Unger *et al.* [21] (figure 1), but without collision detection enabled. Each experimental trial

began after the previous peg was successfully positioned and ended with the successful positioning of the current peg. Participants were asked to position and orientate as quickly as possible a peg into a hole located at the middle of a 3D rectangular parallelepiped. The latter was made transparent to ease the fine positioning of the peg. The peg was made of a rectangular base on which a cylindrical shape was extruded. When both position and orientation were under a given threshold, the peg turned green to indicate it was successfully located. The trial was considered as fully completed when the peg stayed at the correct position for 0.8s. The peg then moved to another position, selected randomly on a hemi-sphere (i.e. the center of the hemi-sphere was the center of the hole and the radius was defined to fit within the display space). The hole remained at the same place. In addition to perspective and occlusion, we also added a ground with shadows projection to improve depth perception. The virtual camera remained fixed during the whole experiment. We controlled for the presence of depth (whether translation along z-axis was required), the combination of axes required for the rotation and the amount of rotation required.

6 males with a mean age of 25 participated. Participants had variable experience with virtual reality and multi-touch displays. 2 were experts, another had some experience, and the others were novices.

6.3 First Results and Discussion

Task completion time is defined as the time it takes to successfully position the current peg into the destination from the last successfully positioned peg. Results exhibited a strong learning effect indicating we should run more than 3 blocks in the final study.

The majority of users feedback concerned *Screen-Space*. They all complained about depth-translation: they were frustrated by being unable to control the depth position with two fingers. They reported they were used to the *pinch-to-zoom* gesture available on commercial products and that handling depth-translation with three fingers was tedious. As our mapping controlled orientation only (i.e. 3 DOF), one extra DOF remained available for the constraint solver (i.e. 2 fingers allow to control up to 4 DOF). We therefore decided to change our two fingers mapping and we added the control of depth-position in addition to rotation (figure 3).

Based on these pilot results, we decided to increase the number of blocks to five in the controlled experiment. We also changed the mapping of two fingers with *Screen-Space* to control both depth-position and orientation.

2. <http://www.immersion.fr>

3. <http://tuio.org>

4. <http://www.ogre3d.org>

5. <https://gist.github.com/764989>

7 CONTROLLED EXPERIMENT

7.1 Goals

The main goal of the experiment is to evaluate the effect of DOF separation on multi-touch displays for 3D manipulation. A second objective is to compare *Sticky Tools* and *Screen-Space* which have never been compared or evaluated.

In designing the experiment, we formulated the following hypothesis :

(H1) Based on the results of the pilot study and user feedback, we hypothesize that separating the control of translation from rotation will increase performance since users will not get confused controlling both at the same time. Thus, *DS3* should be faster.

(H2) Separating the control of translation from rotation increases coordination (in translation or rotation): if users can manipulate DOF for translation and rotation separately in an efficient way, they will be able to improve the coordination of the DOF for the translation or the rotation. *DS3* should have a better coordination.

(H3) The presence of depth-translation will affect performance and coordination, especially with *Sticky Tools* and *Screen-Space* that map depth-translation and rotation together, highlighting the problem pointed out by Nacenta *et al.* [11].

7.2 Task

The task and hardware setup were the same as in the pilot study. However, we changed the design of our application by adding a button that allowed users to skip a trial if they think it was too difficult to complete.

7.3 Participants

10 males and 2 females with a mean age of 24.8 (SD 0.83) participated. 9 were right-handed and 3 were left-handed and all had normal or corrected to normal vision. Participants had a variable experience with virtual reality and 3D applications but this is acceptable as we are observing a lower level physical behaviour. 3 were frequent users of multi-touch displays. 6 participants were familiar with tactile devices such as a tactile mobile phone or tablet-PC, but never worked for a long time on such devices. The other participants had never used a tabletop device.

7.4 Design

A repeated measures design was used. The independent variables were TECHNIQUE, PRESENCE OF

DEPTH, ROTATION LEVEL and ROTATION AMOUNT. There were three levels for TECHNIQUE : *DS3*, *Sticky Tools* and *Screen-Space*. The presentation order of TECHNIQUE was counter-balanced across participants. The PRESENCE OF DEPTH variable had two levels whether depth-position needed to be adjusted. The two levels were NODEPTH and DEPTH. There were also two levels for ROTATION LEVEL, influencing the type of rotation to be performed : SIMPLE and COMPLEX. SIMPLE sets the rotation only around one axis (x , y or z) which was randomly chosen. COMPLEX sets the rotation to be a random mix of x , y and z axes. There were two levels as well for ROTATION AMOUNT, changing the angle of the rotation to perform: SMALL and LARGE. For SMALL, the total amount of rotation was set to 30° and 120° for LARGE.

As suggested by our pilot experiment, we added extra blocks to the experiment. Participants thus completed 5 successive BLOCKS of trials. Each BLOCK consisted of 16 trials: 2 repetitions of 8 PRESENCE OF DEPTH \times ROTATION LEVEL \times ROTATION AMOUNT combinations. The presentation order of TECHNIQUE was counter-balanced across participants. A break was encouraged after each set of 10 trials.

Before starting the experiment with a new technique, participants had a 5 minutes training period to get accustomed to the current technique. The experiment ended with a qualitative feedback from the participants. The experiment lasted approximately 100 minutes in total.

In summary, the experimental design was:

12 participants \times 3 TECHNIQUES \times 5 BLOCKS \times 2 PRESENCE OF DEPTH \times 2 ROTATION LEVEL \times 2 ROTATION AMOUNT \times 2 repetitions = 2,880 total trials

8 RESULTS

8.1 Trial Completion and Number of Touches

Trial completion is defined as the ratio of the number of completed trials to the total number of trials. Trials that were skipped during the experiment were marked as invalid and removed from subsequent analysis.

Repeated measures analyses of variance (ANOVA) found a significant main effect for TECHNIQUE ($F_{2,22} = 7.95$, $p = 0.003$) on trial completion ratio. Pairwise comparisons showed significant differences ($p \leq 0.01$) between *Screen-Space* and the two other techniques. The mean completion ratio was 96.0% for *DS3*, 94.9% for *Sticky Tools* and 86.8% for *Screen-Space*. These results highlight the difficulty of using *Screen-Space* to perform the task.

The number of touches is defined by the total number of *TouchDown* events.

The ANOVA found a significant main effect for TECHNIQUE ($F_{2,22} = 12.54$, $p = 0.002$) on the number of touches. Pairwise comparisons showed significant differences ($p \leq 0.02$) between *Screen-Space* and the two other techniques. The mean number of touches was 12.6 for *DS3*, 17.2 for *Sticky Tools* and 30.0 for *Screen-Space*. We hypothesize that a higher number of touches may be related to a greater difficulty to perform the task as this number is related to the number of iterative steps to accomplish the task.

8.2 Task Completion Time

Task completion time represents the time it takes to successfully position the current peg into the hole from the time it appeared. Skipped trials were removed from the analysis.

The ANOVA found a significant main effect for BLOCK ($F_{4,44} = 4.27$, $p = 0.01$) on task completion time, showing the presence of a learning effect. Pairwise comparisons showed significant differences ($p < 0.05$) between the first block and the others. As a result we removed the first block for subsequent analysis.

The ANOVA also found a significant main effect for TECHNIQUE on task completion time ($F_{2,22} = 17.96$, $p < 0.001$). Pairwise comparisons showed significant differences ($p \leq 0.001$) between *DS3* (11.14s) and the two other techniques, *Sticky Tools* (16.81s) and *Screen-Space* (17.40s). This supports **H1**: *DS3* is 36% faster than *Screen-Space* and 34% faster than *Sticky Tools*.

As expected, the ANOVA revealed a significant main effect of PRESENCE OF DEPTH ($F_{1,11} = 23.37$, $p = 0.001$) on task completion time, decreasing the mean completion time from 16.01s with depth adjustment to 14.14s with no depth adjustment. In addition, ROTATION LEVEL had a significant main effect ($F_{1,11} = 10.37$, $p < 0.01$), reducing the mean completion time from 16.33s with COMPLEX to 13.90s with SIMPLE level. Finally, the ANOVA found a significant main effect of ROTATION AMOUNT ($F_{1,11} = 5.98$, $p = 0.035$), diminishing the mean completion time from 17.40s for LARGE to 12.86s for SMALL level.

8.3 Translation Coordination

The translation coordination ratio is defined as the ratio of the length of the shortest path to the length of the actual path [12].

The ANOVA revealed a significant main effect for TECHNIQUE ($F_{2,22} = 112.19$, $p < 0.001$) on translation coordination. Pairwise comparisons showed significant differences ($p < 0.001$) between *Screen-Space* and

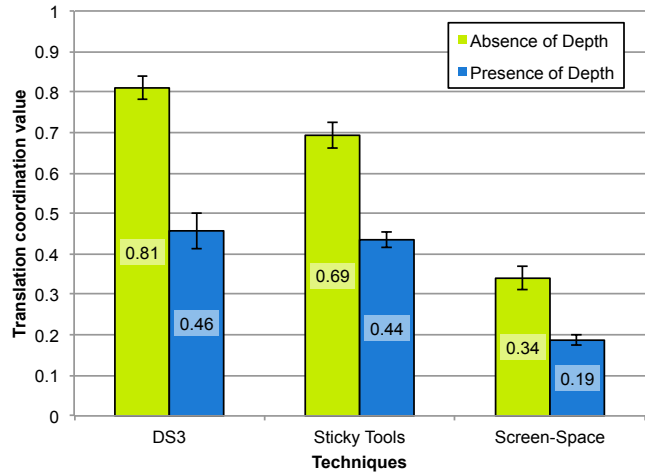


Fig. 5: Mean translation coordination for each technique under the different levels of PRESENCE OF DEPTH. Error bars represent 95% confidence interval.

the two other techniques. Pairwise comparisons also exposed a marginal difference ($p = 0.06$) between *DS3* and *Sticky Tools*. The mean translation coordination was 0.63 for *DS3*, 0.56 for *Sticky Tools* and 0.26 for *Screen-Space*. This result supports **H2**, *DS3* allowing more translation coordination.

As expected (**H3**), the ANOVA also revealed a significant main effect of PRESENCE OF DEPTH ($F_{1,11} = 1286.73$, $p < 0.001$) and a significant TECHNIQUE \times PRESENCE OF DEPTH interaction ($F_{2,22} = 19.81$, $p < 0.001$) on translation coordination. Under the NODEPTH level, *DS3* significantly outperformed *Sticky Tools* ($p = 0.01$) and *Screen-Space* ($p < 0.001$). For this level, the mean translation coordination was 0.81 for *DS3*, 0.69 for *Sticky Tools* and 0.34 for *Screen-Space* (figure 5).

The ANOVA found a significant main effect for ROTATION AMOUNT ($F_{1,11} = 208.25$, $p < 0.001$), with the larger rotation level reducing the mean translation coordination from 0.56 to 0.41. Interestingly, the ANOVA revealed a significant TECHNIQUE \times ROTATION AMOUNT interaction ($F_{2,22} = 17.32$, $p = 0.001$) on translation coordination. Under the SMALL level, *DS3* significantly outperformed *Sticky Tools* ($p < 0.01$) and *Screen-Space* ($p = 0.001$). For this level, mean translation coordination was 0.69 for *DS3*, 0.60 for *Sticky Tools* and 0.40 for *Screen-Space*. The factor ROTATION LEVEL also revealed a significant main effect on translation coordination ($F_{1,11} = 16.49$, $p < 0.010$). Complex rotations reduced the mean translation coordination from 0.52 to 0.46.

Finally, the ANOVA found a significant ROTATION AMOUNT \times PRESENCE OF DEPTH interaction ($F_{1,11} = 23.42$, $p = 0.005$) and a significant TECHNIQUE \times ROTATION AMOUNT \times PRESENCE OF DEPTH interaction

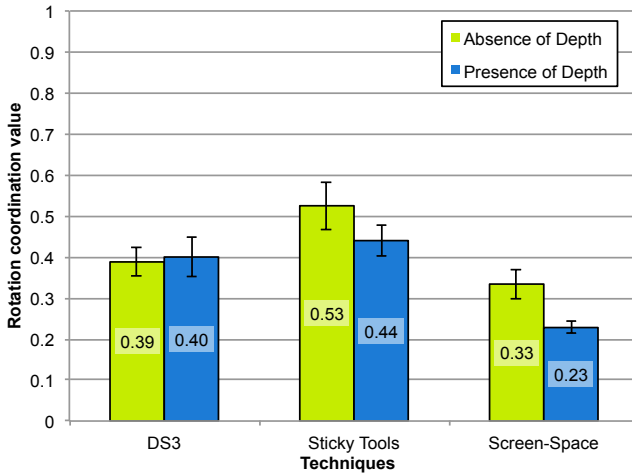


Fig. 6: Mean rotation coordination for each technique under the different levels of PRESENCE OF DEPTH. Error bars represent 95% confidence interval.

($F_{2,22} = 8.93$, $p = 0.006$) on translation coordination. Under the SMALL level of ROTATION AMOUNT and the NODEPTH level of PRESENCE OF DEPTH, DS3 outperformed *Sticky Tools* ($p < 0.001$) and *Screen-Space* ($p = 0.002$). For these levels, mean translation coordination was 0.87 for DS3, 0.74 for *Sticky Tools* and 0.55 for *Screen-Space*.

8.4 Rotation Coordination

The rotation coordination ratio is defined as the ratio of the initial rotation mismatch to the amount of actual rotation [12].

The ANOVA found a significant main effect for TECHNIQUE ($F_{2,22} = 11.71$, $p < 0.005$) on rotation coordination. Pairwise comparisons revealed significant differences between *Sticky Tools* and *Screen-Space* ($p = 0.02$), and between DS3 and *Screen-Space* ($p = 0.02$). The mean rotation coordination was 0.48 for *Sticky Tools*, 0.39 for DS3 and 0.28 for *Screen-Space*. This result supports H2 but not the way we expected. Beyond separating translation DOF from rotation DOF, this indicates that the separation of rotation DOF themselves might lead to better rotation coordination. However, this is just a preliminary result and further investigation would be required in order to conclude.

As hypothesized (H3), the ANOVA found a significant main effect of PRESENCE OF DEPTH on rotation coordination ($F_{1,11} = 12.57$, $p < 0.005$). The mean rotation coordination was 0.48 under the NODEPTH level and 0.39 under the DEPTH level. More interestingly, the ANOVA revealed a significant TECHNIQUE \times PRESENCE OF DEPTH interaction ($F_{2,22} = 11.87$, $p < 0.005$) on rotation coordination. Pairwise comparisons

showed that both *Sticky Tools* and *Screen-Space* significantly reduced their rotation coordination ($p < 0.01$) with the DEPTH level whereas DS3 was not affected ($p = 0.6$). The presence of depth reduced the mean rotation coordination from 0.53 to 0.44 for *Sticky Tools* and from 0.33 to 0.23 for *Screen-Space*. Interestingly, this result shows that PRESENCE OF DEPTH affects only coordination (both rotation and translation) as the ANOVA did not find any significant interaction on mean time.

In addition, the ANOVA found a significant main effect of ROTATION LEVEL on rotation coordination ($F_{1,11} = 33.17$, $p < 0.002$), with complex rotation reducing the mean rotation coordination from 0.44 to 0.33. Interestingly, the ANOVA also found a significant TECHNIQUE \times ROTATION LEVEL interaction ($F_{2,22} = 17.97$, $p < 0.001$) on rotation coordination. Pairwise comparisons found that increasing the complexity of rotation significantly reduced the mean rotation coordination for *Sticky Tools* and *Screen-Space* ($p < 0.001$) whereas DS3 was not affected. Mean rotation coordination decreased from 0.58 to 0.38 for *Sticky Tools* and from 0.34 to 0.22 for *Screen-Space*.

Finally, the ANOVA found a significant PRESENCE OF DEPTH \times ROTATION AMOUNT interaction ($F_{1,11} = 37.91$, $p = 0.002$) and a significant TECHNIQUE \times PRESENCE OF DEPTH \times ROTATION AMOUNT interaction ($F_{2,22} = 7.82$, $p < 0.01$) on rotation coordination. Pairwise comparisons showed that under the SMALL level of ROTATION AMOUNT, changing the level of PRESENCE OF DEPTH significantly reduced the rotation coordination for *Sticky Tools* ($p < 0.01$) and *Screen-Space* ($p = 0.001$) whereas DS3 was not affected ($p = 0.96$). When depth-translation was required, mean rotation coordination decreased from 0.53 to 0.40 for *Sticky Tools* and from 0.41 to 0.19 for *Screen-Space*. This shows that separating translation DOF from rotation DOF helps increasing the rotation coordination (H2).

8.5 DOF Coordination

As explained in the *related work* section, several metrics exist to describe and quantify coordination in docking tasks. In this subsection, we want to understand how the participants used the different techniques to perform the evaluation task. For each technique we want to know how many DOF participants manipulated simultaneously. More precisely we want to determine which DOF were controlled in an integral way and which were controlled in a separable way. Finally, we want to investigate the strategies followed by the participants to adjust the orientation and position.

First, we computed the mean time that users spent in each mode of each technique. To investigate which DOF were manipulated together, we used the NDC

Title	Mean	Title	Mean	Title	Mean
Mode 1d	38%	Mode 1d	42%	Mode 1d	35%
Mode 1d + 1i	7%	Mode 1d + 1i	18%	Mode 2d	33%
Mode 2d	50%	Mode 2d	34%	Mode 3d	32%
Mode 2d + 1i	5%	Mode 2d + 1i	6%		

(a) *DS3* (b) *Sticky Tools* (c) *Screen-Space*

TABLE 1: Mean time spent in each mode

[14]. This metric provides a percentage (Figure 2) of time where participants manipulate n DOF ($1 \leq n \leq 6$). When a single DOF was manipulated, the NDC does not indicate which one actually changed. In this case, the corresponding DOF was identified as the one having the greatest absolute variation in translation or rotation. When several DOF were manipulated together, we used the m -metric [13] to reveal which DOF were controlled simultaneously. Finally, we used the *translation-rotation* coordination ratio defined by Zhai [12] to better understand the strategies employed by participants.

To compute the *translation-rotation* coordination ratio, we first re-sampled each trial into 100 evenly spaced time intervals using linear interpolation. For each interval we then computed the translation and rotation coordination [12], giving a corresponding 2D coordinate. These 100 points provided a representation of the curve starting at coordinate (1,1) (object not orientated nor positioned) and finishing at (0,0) (object well orientated and positioned) (figure 7).

Time spent in modes

The mean time spent in each mode of each technique can be found in table 1. Statistical tests identify influences of independent variables.

For *Screen-Space*, the ANOVA found a significant effect of PRESENCE OF DEPTH for mode 1d ($F_{1,11} = 8.00$, $p = 0.03$) and 2d ($F_{1,11} = 15.40$, $p = 0.011$) but not for mode 3d ($p = 0.55$). When depth manipulation was required, the time spent in mode 1d dropped from 40% to 30% while time spent in mode 2d raised from 29% to 37%. This indicates a trend to use the pinch-to-zoom gesture for depth-translation, as mentioned earlier in the pilot experiment.

For *Sticky Tools*, the ANOVA revealed a significant effect of ROTATION LEVEL only for mode 1d ($F_{1,11} = 30.71$, $p < 0.001$) and 2d ($F_{1,11} = 13.05$, $p = 0.005$). Mode 1d+1i and 2d+1i were not affected (respectively $p = 0.36$ et $p = 0.15$). When complex rotations were required, the use of mode 1d decreased from 46% to 38% while mode 2d increased from 31% to 37%. For *DS3*, we did not find a significant effect of ROTATION LEVEL on the time spent in the different modes. This, combined with rotation coordination result, indicates

Technique	MDS1	MDS2	MDS3	MDS4	MDS5	MDS6
DS3	31%	62%	6.5%	0.5%	0%	0%
Sticky Tools	55%	33%	6.5%	5%	0.5%	0%
Screen-Space	20.5%	14%	17%	42%	6.5%	0%

TABLE 2: Mean NDC values for each technique.

that an integration strategy for controlling rotational DOF is stable no matter the complexity of rotation.

Number of DOF manipulated

The mean NDC for each technique can be found in table 2. The results are consistent with the intended way of integrating and separating DOF for each technique. *Screen-Space* integrates several DOF more often than the two other techniques. To run statistical tests, we regrouped NDC3, 4, 5 and 6 into a global NDC that we named $NDC_{\geq 3}$. We then looked for interactions between TECHNIQUE and independent variables. A TECHNIQUE \times PRESENCE OF DEPTH interaction was found for $NDC_{\geq 3}$ ($F_{2,22} = 6.62$, $p = 0.02$) showing only significant differences for *Sticky Tools* and *DS3* ($p < 0.01$). For *Sticky Tools*, when depth manipulation was required, $NDC_{\geq 3}$ raised from 10% to 16%. For *DS3* we noticed the opposite behaviour: $NDC_{\geq 3}$ dropped from 8% to 6%. Combined with the time spent in mode 2d and the rotation coordination results, this illustrates a consequence of the integration strategy. When depth translation was needed with *Sticky Tools*, users spent significantly more time in mode 2d, controlling significantly more DOF. We believe this conclusion highlights unwanted movements—*DS3*, which separates DOF, seems to avoid this issue.

The ANOVA also found a significant TECHNIQUE \times ROTATION LEVEL interaction ($F_{2,22} = 11.75$, $p < 0.004$) which supports this conclusion. A significant difference was found only for *Sticky Tools* ($p = 0.03$) and *Screen-Space* ($p = 0.006$), not for *DS3* ($p = 0.92$). Complex rotations increased $NDC_{\geq 3}$ from 10% to 16% for *Sticky Tools* and from 61% to 71% for *Screen-Space*.

Single DOF manipulation

Table 3 shows the percentage of time manipulating each individual DOF. Two relevant aspects emerge from the results. First, the manipulation of T_x and T_y is very high, regardless of the technique: the 2 DOF represent more than 50% of single DOF manipulation. Second, this highlights differences in the way the indirect finger is used for different techniques. For *Sticky Tools*, R_x and R_y (controlled by an indirect finger in mode 1d + 1i and 2d + 1i) have high values, due to the use of the indirect finger in a separated way. For *DS3*, T_z (controlled in mode 1d + 1i) presents a

Technique	T_x	T_y	T_z	R_x	R_y	R_z
DS3	8.5%	15.5%	1%	3.5%	1%	1.5%
Sticky Tools	9.5%	20.5%	0%	10.5%	13.5%	1%
Screen-Space	6.5%	13%	0%	0.5%	0.5%	0%

TABLE 3: NDC1 time split for every single DOF.

low value, since this is an integration of this DOF by the user.

Multiple DOF manipulation

We first analyzed subsets of DOF using the m -metric, but were surprised by the results. All subsets presented very low values (< 0.1), except for (T_x, T_y) which stood out (m -metric for $DS3 = 0.36$, for $Sticky Tools = 0.35$ and for $Screen-Space = 0.20$). This reflects the task difficulty which was much more difficult than real object manipulation. Nevertheless, we still wish to know which subsets of DOF were manipulated, and for how long. To calculate these results we used an approach similar to the single DOF measure. Using the results of NDC (e.g. $NDC = n$), we computed the ratio of times the n DOF were manipulated. Table 4 shows the mean results for each technique.

For *Screen-Space* the results are consistent with the taxonomy. The technique tends to integrate more DOF. Combined with the time performance and previous results, we now have insights on the users' strategies. By controlling several DOF at the same time, users cannot perform precise manipulation. Instead, they perform unwanted movements that need to be readjusted later.

For *Sticky Tools*, we can see that the first two subsets include T_z . In addition with the short time spent to manipulate T_z separately, this also supports our previous conclusion regarding unwanted movements. Also, we can see that R_x and R_y are not very integrated, highlighting once again the sequential use of the indirect finger.

For *DS3*, results show an integration of T_z , combined with T_x or T_y , 30% of the time. This gives us some insights regarding the use of the indirect finger. With *DS3*, users manipulate the same type of attribute with both direct and indirect finger (attribute of position), making it possible to combine the action of both fingers. With *Sticky Tools*, the controlled attributes are different (orientation with position) and the two fingers tend to perform more sequentially.

Translation-rotation ratio

Mean translation-rotation ratio can be found in figure 7. For all the techniques, we can see that users start

Couple	Mean	Couple	Mean	Couple	Mean
(T_x, T_z)	20.5%	(T_x, T_z)	11.5%	(T_y, T_z, R_y, R_z)	14.5%
(R_x, R_z)	13%	(T_y, T_z)	6%	(T_y, T_z, R_x, R_z)	14.5%
(R_y, R_z)	13%	(R_x, R_z)	5.5%	(T_x, T_z)	9.5%
(T_x, T_z)	9.5%	(R_y, R_z)	4%	(T_y, T_z, R_x, R_y)	6.5%
(R_x, R_y, R_z)	6.5%	(R_x, R_y)	3.5%	(T_y, T_z, R_x)	6%
(R_x, R_y)	6%	(T_y, T_z, R_x, R_z)	2.5%	(T_y, T_z, R_y)	6%
Other	$\leq 2\%$	Other	$\leq 2\%$	Other	$\leq 2\%$

(a) *DS3*(b) *Sticky Tools*(c) *Screen-Space*

TABLE 4: Time spend manipulating couples of DOF.

to manipulate the position first, then the orientation. The global shape of the curve is similar for *Sticky Tools* and *DS3*, and we can distinguish two phases. First, users primarily manipulate position DOF, reducing translation coordination by 0.6 (i.e. from 1 to 0.4) while reducing rotation coordination by 0.2 (i.e. from 1 to 0.8). Then, they switch to rotation DOF, reducing rotation coordination by 0.8 (i.e. from 0.8 to 0) while reducing translation coordination by 0.4 (i.e. from 0.4 to 0).

For *Screen-Space*, the global shape is different. The manipulation of rotational DOF starts sooner and more generally, the curve is closer to the optimal. This conclusion, together with poor performance, indicates that better coordination does not mean better performance, especially when working on multi-touch table and 3D manipulation.

8.6 Qualitative Feedback

Among the participants, 8 preferred *DS3*, 2 rated *DS3* and *Sticky Tools* equally, and 2 preferred *Sticky Tools*. The participants who said they prefer *DS3* found the technique easier to use and appreciated the fact that rotation was decoupled from translation. Many summarized this saying, this allowed much more accurate control. They also mentioned that performing rotation was easy and felt *natural*. The participant who preferred *Sticky Tools* found the technique less difficult to use for manipulation with both hands. They also reported that they were able to do everything with only one hand. This is something they strongly preferred even if both hands were occasionally required. In contrast to the other participants, they did not like the sequentiality of *DS3*.

Regarding *Sticky Tools*, one user reported that the way to handle rotation was efficient but not intuitive. In addition, they did not like the fact that depth-translation and rotation were linked together. This difficulty came from the coupling of translation and rotation — this was also pointed out by five other participants. Another user reported that the use of an indirect (i.e. not on the 3D object) finger to control orientation was easier to use, in contrast to *DS3* where the external finger controls the depth-position.

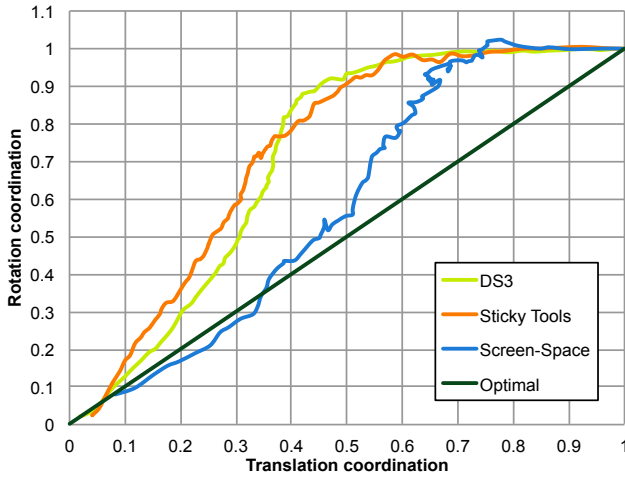


Fig. 7: Mean translation-rotation ratio for each technique. The curve begins at coordinates (1, 1) and ends at coordinates (0, 0). It is function of time (time not represented here).

Regarding the *Screen-Space* technique, all participants reported that this technique was very difficult to use to perform the task. Two participants reported that, although the technique was conceptually intuitive, it was very difficult in practice to perform specific operations. Four participants liked the technique when working on a planar surface such as the top of the peg. They successfully used a gesture highlighted by Reisman *et al.* [9], where 4 fingers manipulate a 3D plane. However, they also pointed out the limitation due to the object size: *Screen-Space* is difficult to use when the size of the object is reduced. Another participant commented that although complex movements were relatively easy to achieve, simple movements were difficult. This was supported by two participants who described the technique as *unpredictable*. Six reported that the integration of all DOF together made the technique difficult to use.

9 DISCUSSION

We designed the experiment to compare three different techniques for performing 3D manipulation tasks on multi-touch displays. *Screen-Space* controlled the six DOF of the task in an integral way, whereas *Sticky Tools* and *DS3* separated the DOF of the task using different strategies.

DOF Separation and Performance

Results show that, for the techniques studied, DOF separation improves performance compared to DOF integration for a docking task on multi-touch displays: *DS3* showed significant lower task completion time

compared to *Screen-Space* and *Sticky Tools*. *DS3* improves performance by 36% compared to *Screen-Space* and by 34% compared to *Sticky Tools*.

This result can be explained by the translation and rotation coordination values showing the effectiveness for controlling the DOF of the task. *Screen-Space*, which tightly couples rotation and translation, revealed the lowest translation and rotation coordination. *Sticky Tools* significantly improves translation coordination by 115% and rotation coordination by 71% compared to *Screen-Space*, while *DS3* improves translation coordination by 142% and rotation coordination by 39% compared to *Screen-Space*.

It appears that for the integral 3D manipulation task we considered, trying to use the separated DOF of a multi-touch display in an integral way provides lower performance. Instead, separating the DOF of the task to match the separated structure of the input device leads to better results. This conclusion extends the work of Veit *et al.* [10] who found similar results for an orientation task.

Jacob's conclusion on integration of DOF was to match as closely as possible the structure of the task with the structure of the input device. But, when working with 2D devices interacting with 3D data, following Jacob's conclusions is not possible. When faced with a choice, our study suggests that performance is improved if the interaction technique follows the structure of the input device rather than the structure of the task.

DOF Separation Strategies

The experiment showed a significant lower task completion time for *DS3* compared to *Sticky Tools* with a 34% improvement for *DS3*. It shows that the strategy of separating DOF can have a severe impact on performance.

Garner [7] showed that orientation and position are two integral attributes of 3D manipulation, making the theory of Jacob *et al.* [5] difficult to apply to multi-touch displays. The lower completion time for *DS3* suggests that orientation and position are still two different attributes which users can easily separate.

In designing *DS3*, we clearly separated translation from the rotation DOF, leading to an inferior completion time but also to a higher translation coordination. Providing the control of orientation and depth-positioning at the same time, *Sticky Tools* has a significantly lower translation coordination compared to *DS3* when no depth-translation was required, highlighting the fact that users had difficulties isolating the control of rotations with *Sticky Tools*. We illustrate this idea in figure 8, which shows the mean *translation-rotation* coordination ratio of a particular participant.

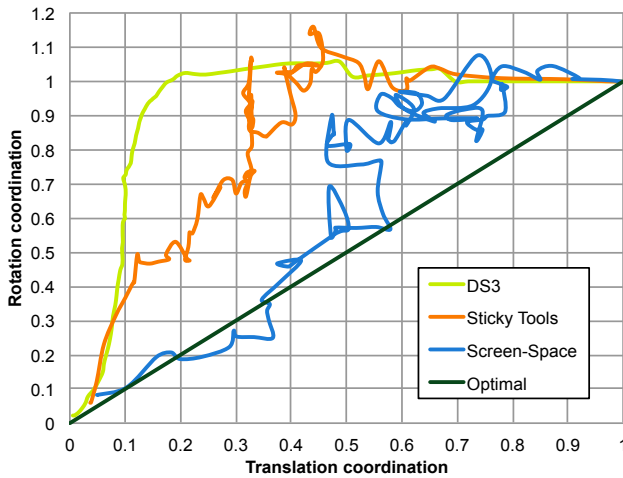


Fig. 8: Typical examples of mean *translation-rotation* coordination ratio from different participants for each technique. It illustrates the integrated control of translation and rotation with both *Sticky Tools* and *Screen-Space* and the separation of translation and rotation with *DS3*.

For *Sticky Tools* and *Screen-Space* we can see that when manipulating rotation, users often changed, even reduced, the coordination in translation.

These conclusions extend the work of Nacenta *et al.* [11] to 3D manipulation. With techniques mapping depth translation and rotation together, it becomes more difficult to perform large rotations without also affecting position, resulting in poor performance and coordination. Introducing *DS3* with the separation of the control of rotation and translation, we increased performance, coordination, and user satisfaction.

These results suggest more separation of DOF. *DS3*, which separates translation DOF, shows the best translation coordination. *Sticky Tools*, which separates rotation DOF, shows the best rotation coordination. A new technique which combines the separation strategy of both *Sticky Tools* and *DS3* would likely improve performance.

The study of DOF coordination also revealed an interesting result, specific to bimanual interaction. Both *Sticky Tools* and *DS3* use an indirect finger, but the techniques differ in the nature of how the DOF are controlled. With *Sticky Tools*, the indirect finger controls an attribute of orientation, while the direct finger handles an attribute of position. For *DS3*, both the direct and indirect fingers control an attribute of position. Our coordination results indicate that controlling the same type of attributes can be parallelized with two hands, while the control of different types of attributes, if mapped to two hands, are performed more sequentially. Designers should consider this when creating two handed interaction techniques for multitouch displays.

Direct vs. Indirect Control

One key point with *DS3* is the mode switching between the control of rotation and translation. In our case, we differentiate rotation from translation according to the number of finger contacts. However, this may have a major drawback when the size of the object being manipulated is small. For example, using three fingers to perform 3D manipulation with *Screen-Space* is difficult with small objects. Indirect control represents a way to solve this issue.

Indirect control provides users with the ability to clearly separate DOF, even with small objects. However, a drawback is the possibility of inadvertently selecting or manipulating another nearby object. When designing interaction techniques for multi-touch displays, interface designers should either prevent interaction with multiple objects at once, or provide clear feedback to show which objects users can interact with.

Another drawback is the manipulation of multiple objects. While *Screen-Space* allows the manipulation of several objects at the same time due to its direct interaction, both *DS3* and *Sticky Tools* use an indirect finger which prevents manipulating multiple objects simultaneously. A way to solve this problem is to provide an area adjacent to each object, dedicated to indirect finger control. In its simple form, this area can be visualized as a circle which appears around or nearby the object being manipulated. Another solution could be to adapt gestures from Rock & Rails interactions [22].

10 CONCLUSION AND FUTURE WORK

We have introduced a taxonomy to classify 3D manipulation techniques for multi-touch displays. The study of 3D interaction techniques in relation to the structure of multi-touch displays led us to introduce *DS3*, a 3D manipulation technique based on a total separation of the position and orientation control. Results of a controlled experiment show the strength of the new technique for both performance and user preference.

This relationship between the separation of DOF and performance confirms recent results showing that the simultaneous manipulation of all DOF does not necessarily lead to the best performance [10]. Our study revealed that the integration of both translation and rotation reduce performance, coordination and user satisfaction. A conclusion which extends the work of Nacenta *et al.* [11], who covered similar issues for 2D manipulation. As future work we wish to explore the design-space introduced by the taxonomy presented in this paper.

ACKNOWLEDGMENTS

This work was supported by the ANR project n° ANR-09-CORD-013 "InSTInCT". We thank the study participants for their time and Daniel Vogel for proof reading.

REFERENCES

- [1] K. Hinckley, J. Tullio, R. Pausch, D. Proffitt, and N. Kassell, "Usability analysis of 3d rotation techniques," in *Proceedings of the 10th annual ACM symposium on User interface software and technology*, 1997, pp. 1–10.
- [2] Y. Wang, C. L. MacKenzie, V. A. Summers, and K. S. Booth, "The structure of object transportation and orientation in human-computer interaction," in *Proceedings of the SIGCHI conference on Human factors in computing systems*, 1998, pp. 312–319.
- [3] J. Rekimoto, "SmartSkin: an infrastructure for freehand manipulation on interactive surfaces," in *Proceedings of the SIGCHI conference on Human factors in computing systems: Changing our world, changing ourselves*, 2002, pp. 113–120.
- [4] M. S. Hancock, S. Carpendale, F. D. Vernier, D. Wigdor, and C. Shen, "Rotation and translation mechanisms for tabletop interaction," in *Proceedings of the First IEEE International Workshop on Horizontal Interactive Human-Computer Systems*, 2006.
- [5] R. J. K. Jacob, L. E. Sibert, D. C. McFarlane, and J. M. P. Mullen, "Integrality and separability of input devices," *ACM Trans. Comput.-Hum. Interact.*, vol. 1, no. 1, pp. 3–26, 1994.
- [6] J. N. Ingram, K. P. Körding, I. S. Howard, and D. M. Wolpert, "The statistics of natural hand movements," *Experimental Brain Research*, vol. 188, no. 2, pp. 223–236, 2008.
- [7] W. Garner, *The Processing of Information and Structure*, 1974.
- [8] M. Hancock, T. T. Cate, and S. Carpendale, "Sticky tools: Full 6dof force-based interaction for multi-touch tables," in *Proceedings of Interactive Tabletops and Surfaces 2009*, 2009.
- [9] J. L. Reisman, P. L. Davidson, and J. Y. Han, "A screen-space formulation for 2d and 3d direct manipulation," in *Proceedings of the 22nd annual ACM symposium on User interface software and technology*, 2009, pp. 69–78.
- [10] M. Veit, A. Capobianco, and D. Bechmann, "Influence of degrees of freedom's manipulation on performances during orientation tasks in virtual reality environments," in *Proceedings of the 16th ACM Symposium on Virtual Reality Software and Technology*, 2009, pp. 51–58.
- [11] M. A. Nacenta, P. Baudisch, H. Benko, and A. Wilson, "Separability of spatial manipulations in multi-touch interfaces," in *Proceedings of Graphics Interface 2009*, 2009, pp. 175–182.
- [12] S. Zhai and P. Milgram, "Quantifying coordination in multiple DOF movement and its application to evaluating 6 DOF input devices," in *Proceedings of the SIGCHI conference on Human factors in computing systems*, 1998, pp. 320–327.
- [13] M. R. Masliah and P. Milgram, "Measuring the allocation of control in a 6 degree-of-freedom docking experiment," in *Proceedings of the SIGCHI conference on Human factors in computing systems*. The Hague, The Netherlands: ACM, 2000, pp. 25–32.
- [14] M. Veit, A. Capobianco, and D. Bechmann, "An experimental analysis of the impact of touch screen interaction techniques for 3-d positioning tasks," in *IEEE Conference On Virtual Reality*, ser. VR '11. New York, NY, USA: ACM, 2011.
- [15] A. D. Wilson, S. Izadi, O. Hilliges, A. Garcia-Mendoza, and D. Kirk, "Bringing physics to the surface," in *Proceedings of the 21st annual ACM symposium on User interface software and technology*, 2008, pp. 67–76.
- [16] O. Hilliges, S. Izadi, A. D. Wilson, S. Hodges, A. Garcia-Mendoza, and A. Butz, "Interactions in the air: adding further depth to interactive tabletops," in *Proceedings of the 22nd annual ACM symposium on User interface software and technology*, 2009, pp. 139–148.
- [17] A. Martinet, G. Casiez, and L. Grisoni, "The design and evaluation of 3d positioning techniques for multi-touch displays," in *3D User Interfaces, 2010 IEEE Symposium*, 20–21 2010, pp. 115–118.
- [18] M. Hancock, S. Carpendale, and A. Cockburn, "Shallow-depth 3d interaction: design and evaluation of one-, two- and three-touch techniques," in *Proceedings of the SIGCHI conference on Human factors in computing systems*, 2007, pp. 1147–1156.
- [19] A. Cohé, F. Dècle, and M. Hachet, "tbox: a 3d transformation widget designed for touch-screens," in *Proceedings of the 2011 annual conference on Human factors in computing systems*, ser. CHI '11, 2011, pp. 3005–3008.
- [20] S. K. Card, J. D. Mackinlay, and G. G. Robertson, "A morphological analysis of the design space of input devices," *ACM Trans. Inf. Syst.*, vol. 9, no. 2, pp. 99–122, 1991.
- [21] B. J. Unger, A. Nicolaidis, A. Thompson, R. L. Klatzky, R. L. Hollis, P. J. Berkelman, and S. Lederman, "Virtual peg-in-hole performance using a 6-dof magnetic levitation haptic device: Comparison with real forces and with visual guidance alone," in *Proceedings of the 10th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, 2002, p. 263.
- [22] D. Wigdor, H. Benko, J. Pella, J. Lombardo, and S. Williams, "Rock & rails: extending multi-touch interactions with shape gestures to enable precise spatial manipulations," in *Proceedings of the 2011 annual conference on Human factors in computing systems*, ser. CHI '11, 2011, pp. 1581–1590.



Anthony Martinet is a PhD student in the MINT research team. His thesis focuses on multitouch interactions, specifically 3-D interactions. Contact him at anthony.martinet@lifl.fr



Géry Casiez is an assistant professor at the University of Lille 1 computer science department and a member of the MINT research team, colabelled by CNRS and INRIA. His research interests include 2D and 3D interaction, haptic interaction, and the empirical evaluation of user interfaces, including associated metrics and predictive models of human performance. Casiez received his PhD in computer science from the University of Lille 1. Contact him at gery.casiez@lifl.fr.



Laurent Grisoni is a professor at the University of Lille 1 computer science department, head of MINT research team, co-labelled by CNRS and INRIA. His research interests include geometric modeling, physical simulation, interactive 3D, and most recently computer-human interaction (gesture-based interfaces). He received his PhD in 1999 from University of Bordeaux I. Contact him at laurent.grisoni@lifl.fr.